CLAIM LISTING

This listing of claims will replace all prior versions, and listings of claims in the application:

AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A system of securely using decryption keys during configuration of an integrated circuit having programmable logic, comprises:

a microcontroller within the integrated circuit for receiving an encrypted bitstream;

a key storage register coupled to the microcontroller for storing key data;

a decryptor coupled to the key storage register, wherein only the decryptor reads from the key storage register; and

a configuration data register in the integrated circuit, wherein the configuration data register cannot be read by the microcontroller after the decryptor is used,

wherein the decryptor is a software decryptor stored in a memory and executed by the microcontroller, wherein the system further comprises hardware, independent of the microcontroller, that ~~selectively~~ allows the microcontroller ~~enables~~ access to the key storage register by unblocking a signal path coupling the microcontroller and the key storage register ~~by allowing the microcontroller access~~ when a program counter of the microcontroller specifies an address within an address range corresponding to the software decryptor within the memory and disallows the microcontroller access to the key storage register by blocking the signal path coupling the microcontroller and the key storage register.

2. (Original) The system of claim 1 wherein the microcontroller stores key data in the key storage register, but the microcontroller cannot read from the key storage register.

3. (Previously Presented) The system of claim 2, wherein the decryptor is a hardware decryptor embedded in the integrated circuit.

4. Cancelled.

5. (Currently Amended) The system of claim [[4]] 1, wherein the memory is a ROM having a decryption engine.

6. (Previously Presented) The system of claim 1, wherein the microcontroller further receives a configuration boot program comprising the decryptor in programmatic form along with the encrypted bitstream comprising encrypted configuration data to be loaded into the configuration data register.

7. (Previously Presented) The system of claim 1, wherein the microcontroller, the key register, the decryptor, and the configuration data register are all within the integrated circuit.

8. (Previously Presented) The system of claim 1, wherein the microcontroller is an emulated microcontroller in the integrated circuit.

9. (Currently Amended) A system of securely using decryption keys during configuration of an integrated circuit having programmable logic, comprising:

a microcontroller within the integrated circuit for receiving an encrypted bitstream;

a key storage register coupled to the microcontroller for storing key data;

a decryption program stored in a memory that uses a predetermined memory address to enable access to the key storage register; and

a configuration data register in the integrated circuit, wherein the configuration data register cannot be read by the microcontroller after the decryption program is used;

wherein access to the key storage register by the microcontroller is allowed only when a program counter of the microcontroller specifies an address within an address range corresponding to the decryption program in the memory by unblocking a signal path coupling the microcontroller and the key storage register,

wherein access to the key storage register by the microcontroller is disallowed when the program counter of the microcontroller specifies an address outside of an address range corresponding to the decryption program in the memory by blocking the signal path coupling the microcontroller and the key storage register.

10. (Original) The system of claim 9, wherein the memory is a ROM containing a decryption engine.

11. (Original) The system of claim 9, wherein the microcontroller further receives a configuration boot program along with the encrypted bitstream.

12. (Currently Amended) A method of securely using decryption keys during ~~field programmable gate array~~ configuration of an integrated circuit comprising programmable logic, comprising the steps of:

receiving an encrypted bitstream at a microcontroller within the field programmable gate array;

loading a decryptor with data from a key register;

loading the decryptor with data from the microcontroller; [[and]]

loading a configuration data register with a decrypted bitstream from the decryptor, wherein the configuration data register cannot be read by the microcontroller after the decryptor is used; [[and]]

~~selectively~~ enabling access to the key register by unblocking a signal path coupling ~~allowing~~ the microcontroller and the key register ~~access~~ only when a program counter of the microcontroller specifies an address within an address range of the decryptor; and

disabling access to the key register by blocking the signal path coupling the microcontroller and the key register when the program counter of the microcontroller specifies an address outside of the address range of the decyptor.

13. (Original) The method of claim 12, wherein the method further comprises the step of loading the key register with key data from the microcontroller.

14. (Previously Presented) The method of claim 12, wherein the configuration data register cannot be read by the microcontroller while the decryptor is used.

15. (Original) The method of claim 12, wherein the microcontroller cannot read from the key register.

16. (Currently Amended) The method of claim 12, wherein only the decryptor can read from the key ~~storage~~ register.

17. Cancelled.

18. (Currently Amended) A system of securely using decryption keys during ~~programmable logic device~~ configuration of an integrated circuit comprising programmable logic, comprises:

a memory-mapped key register coupled to a microcontroller data bus;

a decryptor engine stored in non-volatile memory and coupled to the microcontroller data bus; and

logic circuitry limiting access to the key register from the microcontroller data bus using specified addresses of the non-volatile memory corresponding to the decryptor engine and a received program counter value of a microcontroller,

wherein the logic circuitry selectively blocks or unblocks a signal path coupling the microcontroller data bus and the key register according to the specified addresses of the non-volatile memory corresponding to the decryptor engine and the program counter value.

19. (Previously Presented) The system of claim 18, wherein the logic circuitry uses specified addresses of the non-volatile memory by limiting access to minimum and maximum ROM memory addresses using the microcontroller program counter.

20. (Currently Amended) A computer-readable medium comprising instructions written thereon in the form of a bitstream that configures an integrated circuit comprising programmable logic ~~a programmable logic device~~, the computer-readable medium comprising:

a configuration boot program portion of the bitstream that runs a microcontroller on the integrated circuit ~~programmable logic device~~; and

an encrypted bitstream portion of the bitstream containing encrypted configuration data that when decrypted and loaded into a configuration data register on the integrated circuit ~~programmable logic device~~ configures the programmable logic ~~device~~,

wherein the configuration boot program further comprises instructions for a decryptor, wherein the configuration boot program stores the instructions for the decryptor in a memory, wherein the decryptor is executed by a microcontroller and decrypts the encrypted bitstream using key data stored within a key storage register, and wherein access to the key storage register by the microcontroller is selectively permitted by blocking or unblocking a signal path coupling the microcontroller to the key storage register according to whether ~~when~~ a program counter of the microcontroller specifies an address within an address range corresponding to the decryptor within the memory.

21. (Cancelled)

22. (Previously Presented) The computer-readable medium of claim 20, wherein the configuration boot program comprises instructions for a decompressor.